

Коспа таълим

2-курс

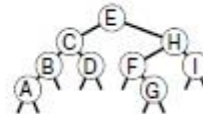
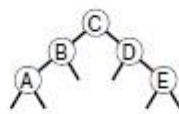
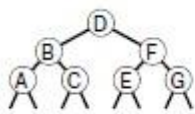
“Алгоритмы и структура данных”

1. Основные понятия и определения алгоритмов и структур данных.
2. AVL-дерево. Структура дерева и операции над его элементами
3. Создайте Алгоритм Евклида
4. Сложные типы данных. Работа с записями, структурами, классами
5. Красно-чёрное дерево. Структура дерева и операции над его элементами
6. Напишите рекурсивную программу для вычисления $\lg(N!)$.
7. Типы значений и ссылочные типы данных.
8. Код Хаффмана
9. Однонаправленный связный список.
10. Метод закрытого хеширования
11. Метод открытого хеширования
12. B⁺-дерево. Структура дерева и операции над его элементами
13. B-дерево. Структура дерева и операции над его элементами
14. Структура списка и операции над его элементами.
15. Хешированные и индексированные файлы
16. Двухнаправленный связный список.
17. Особенности работы с внешней памятью.
18. Структура списка и операции над его элементами.
19. Неориентированные графы.
20. Очередь. Структура очереди и операции над её элементами
21. Кодирование длин серий (RLE)
22. Работа со значимыми и ссылочными переменными.
23. Стек. Структура стека и операции над его элементами
24. Сильно-связный оргграф. Алгоритмы поиска компонент сильной связности в орграфе
25. Инфиксные, префиксные, постфиксные выражения.
26. Алгоритмы на орграфах
27. Алгоритмы преобразования и вычисления выражений
28. Приведите последовательности значений аргумента, получаемых в результате вызова программы 5.2 для каждого из целых чисел от 1 до 9.
29. Определение, терминология и способы представления орграфа
30. Дерево. Общие сведения и терминология.
31. Определение и построение бинарного дерева поиска
32. Код Хаффмана
33. Операции над бинарным деревом поиска. Обходы в глубину и ширину
34. Сортировка извлечением

35. Кодирование длин серий (RLE)
36. Создайте нерекурсивную реализацию алгоритма Евклида.
37. Нахождение минимального остовного дерева в неориентированном графе с использованием алгоритмов Прима, Краскала.
38. Определите глубину рекурсии алгоритма Евклида для двух последовательных чисел Фибоначчи (F_N и F_{N+1}).
39. Напишите рекурсивную программу для вычисления постфиксных выражений.
40. Напишите рекурсивную программу для вычисления инфиксных выражений, в которых операнды всегда заключены в круглые скобки.
41. Напишите рекурсивную программу, которая преобразует инфиксные выражения в постфиксные.
42. Напишите рекурсивную программу, которая преобразует постфиксные выражения в инфиксные.
43. Алгоритмы нахождения минимального остовного дерева в неориентированном графе
44. Напишите рекурсивную программу, которая находит максимальный элемент в массиве, выполняя сравнение первого элемента с максимальным элементом остальной части массива (найденным рекурсивно).
45. Напишите рекурсивную программу, которая находит максимальный элемент в связном списке.
46. Методом индукции докажите, что количество вызовов функции, выполняемых любым алгоритмом "разделяй и властвуй", который делит задачу на части, в сумме составляющие задачу в целом, а затем решает части рекурсивно, линейно относительно размера задачи.
47. Докажите, что рекурсивное решение задачи о ханойских башнях является оптимальным. То есть покажите, что любое решение требует по меньшей мере $2N - 1$ перекладываний.
48. Напишите рекурсивную программу, которая вычисляет длину i -ой метки на линейке с $2^n - 1$ метками.
49. Алгоритмы сжатия данных семейства Лемпеля-Зива (LZ*)
50. Напишите рекурсивную программу, которая заполняет массив размером $n \times 2^n$ нулями и единицами таким образом, чтобы массив представлял все n -разрядные числа
51. Докажите следующее свойство программы рисования линейки: если разность между ее первыми двумя аргументами является степенью 2, то оба ее рекурсивных вызова также обладают этим свойством.
52. Напишите функцию, которая эффективно вычисляет количество завершающих нулей в двоичном представлении целого числа.

53. Нахождение минимального остовного дерева в неориентированном графе с использованием алгоритмов обратного удаления, Борувки.
54. Допустим, деревья содержат элементы, для которых определена операция \equiv . Напишите рекурсивную программу, которая удаляет в бинарном дереве все листья, содержащие элементы, равные данному
55. Сортировка Шелла
56. Поиск компонент сильной связности в орграфе с использованием алгоритмов Косараджу, Тарьяна.
57. Упорядоченные деревья эквивалентны согласованным строкам из пар круглых скобок: упорядоченное дерево - это либо пустая строка, либо последовательность упорядоченных деревьев, заключенных в круглые скобки. Нарисуйте упорядоченное дерево, соответствующее строке
 $((()((()())())((()())())))$
58. Сортировка включением
59. Нахождение центра ориентированного графа.
60. Напишите программу для определения того, являются ли два бинарных дерева изоморфными неупорядоченными деревьями.
61. Сортировки подсчетом
62. Поиск кратчайших путей в орграфе с использованием алгоритмов Дейкстры.
63. Нарисуйте все упорядоченные деревья, которые могли бы представлять дерево, определенное набором ребер $0-1, 1-2, 1-3, 1-4, 4-5$.
64. Алгоритм КМП
65. Поиск кратчайших путей в орграфе с использованием алгоритмов Беллмана-Форда.
66. Докажите, что если в связном графе, состоящем из N узлов, удаление любого ребра влечет за собой разъединение графа, то в нем $N - 1$ ребер и ни одного цикла.
67. Анализ алгоритмов
68. Поиск кратчайших путей в орграфе с использованием алгоритмов Флойда-Уоршелла.
69. Сколько внешних узлов существует в M -арном дереве с N внутренними узлами? Используйте свой ответ для определения объема памяти, необходимого для представления такого дерева, если считать, что каждая ссылка и каждый элемент занимает одно слово памяти.
70. Свойства алгоритмов
71. Реализация алгоритмов сжатия данных семейства LZ^* ($LZ77, LZ78, LZW$).
72. Приведите верхнее и нижнее граничные значения высоты M -арного дерева с N внутренними узлами.
73. Виды алгоритмов

74. Реализация алгоритма сжатия данных RLE.
75. Приведите верхнее и нижнее граничные значения длины внутреннего пути M-арного дерева с N внутренними узлами.
76. Стеки и очереди
77. Реализация алгоритма сжатия данных Хаффмана.
78. Приведите верхнее и нижнее граничные значения количества листьев в бинарном дереве с N узлами.
79. Хэш таблицы
80. Прямой, симметричный, обратный обходы дерева.
81. Приведите порядок посещения узлов для прямого, поперечного, обратного и обхода по уровням для следующих бинарных деревьев:



- 82.
83. Сортировки
84. Операции поиска, включения, удаления записи из бинарного дерева поиска.
85. Покажите, что прямой обход леса эквивалентен прямому обходу соответствующего бинарного дерева а обратный обход леса эквивалентен поперечному обходу бинарного дерева.
86. Поиск кратчайшего пути
87. Алгоритм обхода дерева в ширину с использованием очереди.
88. Приведите нерекурсивную реализацию поперечного обхода.
89. Взвешенные графы
90. Определение основных понятий, относящихся к структуре «дерево».
91. Приведите нерекурсивную реализацию обратного обхода.
92. Пирамидальная сортировка
93. Хэш функции
94. Напишите программу, которая принимает на входе прямой и поперечный обходы бинарного дерева и генерирует обход дерева по уровням.
95. Быстрая сортировка
96. Определение B-дерево
97. Напишите программу, которая подсчитывает листья в бинарном дереве.
98. Поиск заданного ключа
99. Таблицы с прямой адресации
100. Напишите программу, которая подсчитывает количество узлов в бинарном дереве с одним внешним и одним внутренним дочерними узлами.